

Method and Apparatus for recording a data stream on a storage medium

The invention relates to a method and an apparatus for
5 recording a data stream on a storage medium, especially for recording on write once-media.

Background

10 A write-once optical disk known under the acronym CD-R is broadly used in computer technology and is nowadays implemented in most of the current PCs and notebooks. However, data amounts to be stored are increasing continuously, resulting in the demand for higher storage
15 capacity. Rewritable DVD's (DVD-RAM, DVD-RW, DVD+RW) are on the market, having a storage capacity of 4.7 GB. Furthermore, a next generation of large capacity optical discs called "Blu-ray Disc" has recently been announced, enabling the recording, rewriting and play back of up to 27
20 GB of data. This allows e.g. archiving computer data with high capacity or recording of two hours high-definition television video.

Defects occurring during the writing process have to be
25 addressed in order to avoid re-writing the whole data, resulting in the case of write-once media in the use of a further disk. This is especially important for realtime recording of audio-visual (AV) data streams on write-once media or rewritable media and realtime playback of such
30 recorded streams, since defects would result in very annoying interrupts of the picture or audio signal.

Invention

A problem to be solved by the invention is to enhance the resistance against defects during recording data on a storage medium and playing back the recorded data.

This problem is solved by the recording method disclosed in claim 1 and by the play back method disclosed in claim 8. An apparatus that utilizes this method is disclosed in claim 13.

According to the invention a data stream is recorded in data blocks on a storage medium, wherein the following steps are performed:

- generating an error correction block for one or more data blocks;
- writing said error correction block on said storage medium during recording;
- keeping a spare data area on the storage medium blank;
- reconstructing a defect data block using said error correction block; and
- storing said reconstructed data block in said spare data area.

Advantageous additional embodiments of the invention are disclosed in the respective dependent claims, the following description and the figures.

Drawings

Exemplary embodiments of the invention are described with reference to the accompanying drawings, which show in:

Fig. 1 an optical disc with a currently recorded track;

Fig. 2 the spiral structure of the disc track containing payload blocks, parity blocks and local spare area blocks;

5 Fig. 3 schematically the write process of a data stream on the disc track;

Fig. 4 schematically the write process of a stream with an error occurring during writing;

10

Fig. 5 the verifying process of the recorded data stream depicted in fig.4;

Fig. 6 the default playback process of a recorded stream;

15

Fig. 7 an enhanced playback process of a recorded stream for a pick-up having additional memory;

Fig. 8 an enhanced playback process of a recorded stream for a pick-up with just in time reconstructing;

20

Fig. 9 a method to use two different parity blocks to handle overlapping defects;

25 Fig. 10 schematically a block sequence with two different parity blocks and two neighbouring defected data blocks;

Fig. 11 schematically a method using additional parity blocks covering several block sequences;

30

Fig. 12 an example of defected data blocks and parity blocks which can be reconstructed using the method shown in fig. 11.

35

Exemplary embodiments

Figure 1 shows an example of an optical disc with a currently recorded track T. The area A around this circle (of course, it's not a real circle; it's one rotation of the track spiral) is the area for a fast jump, i.e. all tracks covered by this area can be fast accessed by the optical pickup, i.e. the pick-up is able to jump to nearly each part of this area within one disc rotation. Furthermore, a defect block D, spare areas S and a parity block P are depicted.

The parity block P is inserted during recording on the disk and comprises data to reconstruct single defect payload blocks. This is a preventive step to provide additional information in order to reconstruct defect blocks.

More specifically, during the disk write the file system computes the parity block by e.g. XOR (logical exclusive or) combination of the corresponding bits in the covered data blocks. If later on the disk controller gives a bad block error, the file system or the pick-up itself regenerates the defect block from the other blocks and the parity block.

The parity blocks may cover one or more sequences of data blocks. Furthermore, different levels of parity blocks may be possible, i.e. some parity blocks cover only payload blocks. Other parity blocks may cover more blocks including other parity blocks. This is described in more detail below.

The use of parity blocks has several advantages:

- The method is widely scalable, wherein the manufacturer/implementer may define the number of parity blocks and the number of covered payload blocks by a parity block.
- The recording itself can be performed in realtime by using an 1x drive.

- A reconstruction in realtime during reading can be performed by using an additional buffer plus an additional logic to regenerate defect blocks by using the parity block.

- 5 Example parameters for parity blocks may be 1 ECC block as parity block for 31 payload ECC blocks, wherein ECC (Error Correction Code) is known to be used in various forms for CD, DVD, Blu-ray disks.
- 10 During recording of the stream the recording skips some blocks. These blocks are reserved as spare areas S. After recording a post processing investigates all the defect blocks, reconstructs these bad blocks and stores the reconstructed blocks in such spare areas.
- 15 The spare areas are provided in the near area of each track in order to allow replacement blocks with short jumps or even without jumps by buffering these spare areas during playback.
- 20 Advantageously, spare areas are scattered over the whole disc by rotation angle and radius. This avoids situations, where one single spot could destroy a lot of spare areas.
- 25 Furthermore, the spare areas are advantageously scattered over the disc radius in a way, that the pick-up arrives and leaves such a spare area from each point of the disc within one single rotation. That requires,
- that the spare area is in a neighborhood track and
 - 30 • not parallel to the defect block, i.e. at another angle than the defect block, wherein a good angle is approximately 180 degree.

Advantageously, the used spare area can be located before
35 the bad block. The optical disk drive buffers the blocks of the read spare areas during playback. When the defect block

is derived for which the buffered block is the replacement the optical disk drive can deliver this buffered block immediately without any jump back. This reduces jump backs, but requires more buffer to keep read replacement blocks.

- 5 After delivering such a replacement block the RAM for this buffered block can be reused for new read replacement blocks.

Advantages of the use of spare areas are:

- 10 - Due to the location of spare areas near the defect blocks a replacement of defect blocks even during realtime playback is performable.
- When the spare areas are filled offline, i.e. after recording of the stream, the realtime recording can be
- 15 performed by a 1x drive.
- Days after the recording, the disc may be locally damaged (e.g. by fingerprints). The new defect blocks can be reconstructed by using the parity blocks. The reconstructed blocks can be stored in these local spare areas. So, the
- 20 robustness of the data is increased, even by defects generated after the recording session. I.e. this method provides a post protection and restoring of the payload by providing realtime playback.
- 25 An example definition for such spare areas is that a spare area consist of 16 ECC blocks and the distance between 2 spare areas shall be 10 and nearly a half disc rotations.

Figure 2 depicts the spiral structure of the disc track, wherein only a part of the track is shown. The track is read and write from the inner start of the spiral to the outer end of the spiral spiral (or the other way around, e.g. for the second layer of a Blu-ray disc). The spiral contains the blocks. These blocks are of the type payload block

30 B0,...,B7, parity block P or local spare area SA0 and SA1.

35 In this example there is a sequence of 8 payload blocks

B0,...,B7, followed by a parity block P. This parity block P contains the parity bits for the previous 8 payload blocks B0,...,B7, meaning that it is possible to reconstruct one defect payload or parity block. The combination of the
5 payload blocks B0,...,B7 together with its parity block P are called in the following a PP sequence (PP=payload and parity). Sometimes two local spare area blocks SA0, SA1 occur immediately after a parity block P. One local area and the next following local spare area are located at different
10 angles, the closer to 180 degree the better.

Figure 3 describes the write process of a stream. Steps R1 to R6 describe a part of this writing process:

- 15 R1: Writing of 8 payload blocks B0, ..., B7.
- R2: Writing the parity block P of the just written 8 payload blocks B0, ..., B7.
- R3: Keep space for 2 blocks inside the local spare area:
Writing of a run-out; then skip the space of two blocks,
20 i.e. keep unrecorded space for the blocks SA0, SA1.
- R4: Writing of a run-in to continue the recording. Writing of the next 8 payload blocks B0, ..., B7.
- R5: Writing of the parity block P of the just written 8 payload blocks B0, ..., B7.
- 25 R6: Writing of the next 8 payload blocks B0, ..., B7, i.e. this time no space is saved for a local spare area.

Figure 4 describes the write process of a stream, wherein an error occurs during writing. Steps RD4 to RD6 describe a
30 part of this writing process:

- RD4: Writing of the next 8 payload blocks B0, ..., B7. The writing of the last payload block DB7 fails. The pick-up is not able to detect this error during recording, i.e. the
35 device need a verify after recording of the whole stream to detect this bad block.

RD5: Writing of the parity block P of the just written 8 payload blocks B0, ..., B7. This parity block is required to reconstruct DB7 during a verify after the recording of the whole stream.

5 RD6: Writing of the next 8 payload blocks B0, ..., B7.

Figure 5 describes the verify process of a recorded stream. The complete recording is scanned during the verify process. In this example the defect block DB7 must be recognized,
10 reconstructed and replaced. Steps V1 to V6 describe a part of this verify process:

V1: After reaching DB7 the pick-up recognizes that DB7 is a defect payload block. Each correct read block of this PP
15 sequence is combined with the previously correct read block by applying a logical exclusive-OR operation, i.e. one block in the pick-up internal RAM (EOR block) is used for this concern.

V2: The pick-up reads the parity block of this PP sequence
20 and reconstructs DB7 inside the pick-up internal RAM by using the EOR block.

V3: Jump back to a local spare area, which is located at an angle ASADB7 far away from DB7, e.g. at an angle ASADB7 between 90 to 270 degree.

25 V4: Skip all the blocks until arriving the local spare area. Write the reconstructed payload block B7 into the next free local spare area block SA0, including run-in and run-out.

V5: Skip all the blocks until arriving the next payload block B0, which is not yet verified.

30 V6: Continue verify.

Figure 6 describes the default playback process of a recorded stream. In this example the defect block DB7 must be recognized and recovered. Steps PBa to PBc and PB1 to PB5
35 describe parts of this playback process:

PBa to PBc:

PBa: Read the payload blocks until the parity block of this PP sequence.

PBb: Skip the parity block.

PBc: Continue the reading of the payload blocks.

5 PB1 to PB5:

PB1: Read the payload blocks until the B6 block.

PB2: B7 is marked in the defect list as defect. Jump back in the track before, i.e. jump to a track position just before the replacement block SA0/B7. Skip all blocks before the replacement block SA0/B7.

PB3: Read the replacement block SA0/B7 which is the replacement block of the defect payload block DB7.

PB4: Skip all the already read blocks, i.e. skip all the blocks until the parity block just after DB7.

15 PB5: Continue the reading of the payload blocks B0....

Figure 7 describes an enhanced playback process of a recorded stream. The pick-up has additional memory available to buffer replacement blocks of local spare areas. In this example the defect block DB7 must be recognized and recovered. Steps PBB1 to PBB5 describe a part of this playback process:

PBB1: Read the payload blocks and skip the parity block of this PP sequence.

PBB2: There is one replacement block B7/SA0 inside the local spare area. Read this replacement block and store it in the additional buffer.

PBB3: Skip SA1 of this local spare area. Read the payload blocks until B6.

PBB4: Recover DB7 by using the buffered payload block B7/SA0. Free the memory of the additional buffer. Skip all the blocks, until the next not yet read payload block B0.

PBB5: Continue the reading of the payload blocks B0... .

35

Figure 8 describes a pick-up for just in time

reconstructing. This allows to reconstruct even not yet recognized defect payload blocks during playback, i.e. just in time. This feature requires a buffer to cover all payload blocks of a PP sequence for the case that the first payload
5 block of a PP sequence is defect. In this example the defect block SDB6 must be recognized and recovered. Steps PBP1 to PBP7 describe a part of this playback process:

PBP1: Read the payload blocks B0 to B7 into the buffer and
10 send the read blocks to the device.

PBP2: Skip the parity block of this PP sequence and the following local spare area. Note: in this example no replacement block is inside the local area.

PBP3: Read the payload blocks B0 to B5 into the buffer, i.e.
15 overwrite the old buffer content, and send the read blocks to the device.

PBP4: B6 cannot be read: A spontaneous defect block SDB6 is detected. Skip this block SDB6.

PBP5: Continue the reading of the payload block B7 into the
20 buffer, but do not send this block to the device yet, because B6 must be sent first.

PBP6: Read the parity block. Reconstruct the defect payload block SDB6/B6 of this PP sequence by using the buffered blocks B0 to B5, B7 and the parity block P from the buffer.
25 Send the remaining payload blocks B6 and B7 to the device.

PBP7: Continue playback by reading the next PP sequence.

The most enhanced pick-up version, i.e. max. robustness by min. pick-up speed requirement, would be a pick-up, which
30 buffers a whole PP sequence and several local spare areas, i.e. a combination of figure 7 and 8.

Figure 9 describes a method to use two different parity blocks PO, PE to handle overlapping defects, e.g. a defect
35 of B1 and B2 as shown in figure 10. Even a reconstruction of B15 and PE is possible.

The first parity block PE is generated for the blocks B0, B2, ..., B14, the second parity block PO for the blocks B1, B3, ..., B15.

5

Figure 11 shows schematically a method using additional parity blocks covering several block sequences, which further increases the robustness in comparison to figure 9 and allows the reconstruction of up to four contiguous defect blocks as shown in figure 12. The additional parity blocks PGE and PGO cover the last four PP sequences. Each single block excluding PGE and PGO can be reconstructed twice:

1. by the PP sequence internal PE and PO and
- 15 2. by the PGE and PGO.

In this example, the additional required memory to add PE and PO is 12.5 %. But, the additional required memory to add PGE and PGO is only less than 2.8 %.

20 The invention is especially useful for recording of an AV data stream on a write once-media optical disk. But, it also increases the robustness of recorded AV streams on rewritable discs. However, it is also applicable to any other kind of data storage.

25

In order to recognize parity clusters and local spare area clusters these cluster types shall be indicated by the DFL (Defect List; see 6.6.5.2, Blu-ray Spec, part 1, version 1.0). A parity cluster or a contiguous parity cluster sequence (e.g. PE and PO) shall be indicated by a special Status 1 value in the DFL Entry: e.g. 1001, type parity. The Defective PSN field identifies the address of the first parity cluster of a parity cluster sequence (first parity cluster of a PP sequence). The Replacement PSN field shall indicate the number of parity clusters (e.g. 2 if PE and PO exists).

35